

Практическое занятие № 3

Соединение таблиц

Цель занятия. Научиться выполнять соединение таблиц, освоить различные варианты соединения: естественное (внутреннее), неестественное, внешнее.

В реляционной БД взаимосвязанная информация в ходе нормализации распределяется по отдельным таблицам. Соединение позволяет объединить данные связанных таблиц.

Различают следующие виды соединения:

- естественное;
- неестественное;
- внешнее соединение

Виды соединений рассмотрим на примере БД Поставки (рис.1). Данные таблиц БД приведены в [практическом занятии 1](#).



Рис. 1 База данных Поставки

Для иллюстрации всех видов соединений в БД были внесены изменения. В таблице book (Книга поставок изменена колонка Код покупателя (разрешено содержать значения null). Это изменение можно внести оператором

```
ALTER TABLE book  
    modify post_kod integer NULL
```

Затем в любой строке поставок (таблица book) очистим колонку post_kod. Для определенности выдерем поставку с идентификатором поставки book_id=3.

Если же данные генерируются в операторе WITH, достаточно в данных для таблицы book в строке с идентификатором book_id=3 значение колонки post_kod необходимо заменить на null (рис.2).

	book_id	ndoc	dat	post_kod
1	1	10	2023-01-30	1
2	2	11	2023-01-31	3
3	3	16	2023-02-02	(NULL)
4	4	19	2023-02-04	1
5	5	21	2023-02-05	4
6	6	26	2023-02-09	5
7	7	28	2023-02-10	7
8	8	29	2023-02-10	2

Рис. 2 Таблица книги поставок после изменений

1) Естественное соединение (внутреннее)

Естественное соединение использует существующие в БД связи таблиц Primary Key(PK) - Foreign Key (FK).

Пример 1. Выберем все поставки (номер документа, дату поставки) с указанием наименования поставщика. Для получения указанных данных необходимо выполнить соединение двух таблиц: *Книги поставок (book)* и *Поставщиков (spost)*, которые имеют связь

PK (spost.post_kod) – FK(book.post_kod).

Составим оператор запроса и включим в раздел WHERE предикат сравнения:

```
SELECT book_id, ndoc, dat, b.post_kod, post_name
FROM spost sp, book b
WHERE sp.post_kod=b.post_kod
```

Алгоритм выполнения: После прямого произведения таблиц в разделе FROM из всех возможных комбинаций строк таблиц предикатом сравнения раздела WHERE отбираются те, которые совпадают по колонке *Код поставщика (post_kod)*. Предикат сравнения задает условие естественного соединения таблиц.

Результат запроса (рис. 3) показывает, что поставка с book_id=3 не вошла в результирующий набор, т.к. она не удовлетворяет условию соединения таблиц (ей не нашлась пара в таблице поставщиков).

	book_id	ndoc	dat	post_kod	post_name
1	1	10	2023-01-30	1	АО Продлог
2	2	11	2023-01-31	3	ООО Логистик
3	4	19	2023-02-04	1	АО Продлог
4	5	21	2023-02-05	4	ООО Москворечье
5	6	26	2023-02-09	5	ОАО Черемушки
6	7	28	2023-02-10	7	АО Автолог
7	8	29	2023-02-10	2	ООО Контакт

Рис. 3 Результат соединения книги поставок с таблицей поставщиков

Предикат сравнения в разделе WHERE можно заменить оператором JOIN в разделе FROM:

ГлавнаяТаблица JOIN ПодчиненнаяТаблица

ON ГлавнаяТаблица.FK=ПодчиненнаяТаблица.FK

Часто указывают полный вариант записи соединения (его еще называют внутренним) INNER JOIN, которое дословно переводится как внутреннее соединение, в отличие от внешних соединений, рассматриваемых далее.

Изменим оператор запроса примера 1:

```
SELECT book_id, ndoc, dat, b.post_kod, post_name
FROM spost sp JOIN book b ON
sp.post_kod=b.post_kod
```

Условие соединения таблиц может сочетаться с другими условиями.

Пример 2. Выбрать все поставки продукции с кодом 1: поставщик, дата поставки, количество, цена, стоимость.

В данном примере нам необходимо соединить уже 3 таблицы: *Поставщики* (spost) с *Книгой поставок* (book) по колонке *Код поставщика* (post_kod) и полученные строки соединить с *Ассортиментом поставки* (kart) по колонке *Идентификатор поставки* (book_id). И предикатом сравнения отобрать строки, относящиеся к заданной продукции.

```
SELECT post_name, dat, kol, cena,
       cast(kol*cena as Numeric(10,2)) stoim
FROM spost sp JOIN book b ON sp.post_kod=b.post_kod
JOIN kart k ON b.book_id=k.book_id
WHERE prod_kod=1
```

Результат запроса:

	post_name	dat	kol	cena	stoim
1	АО Продлог	2023-01-30	10	45.0	450.00
2	ООО Логистик	2023-01-31	12	45.0	540.00
3	ООО Москворечье	2023-02-05	6	25.5	153.00

Упражнение 1. Выбрать поставки за февраль 2023 г.: поставщик, дата поставки, код продукции, наименование продукции, цена, количество

2) Неестественное соединение

Этот вид соединения основывается на данных, а не на структуре БД (связях между таблицами).

Пример 3. Отобрать все пары поставщиков таким образом, чтобы пару составляли поставщики из одного города.

Так как надо образовать пары, то необходимо соединить таблицу *Поставщик* с ее копией. Далее из образованных пар поставщиков выбрать те, в которых совпадает город. При этом необходимо использовать **алиасы (псевдонимы)**, которые позволяют различать соединяемые копии таблиц. Используем в качестве псевдонимов короткие имена p1 и p2.

```
SELECT p1.city, p1.post_name, p2.post_name
FROM spost p1, spost p2
WHERE p1.city=p2.city
```

Результат запроса (первые несколько строк)

	city	post_name	post_name	
1	Барнаул	ООО Логистик	ООО Логистик	пары - близнецы
2	Москва	АО Автолог	АО Автолог	
3	Москва	ООО Контакт	АО Автолог	зеркальные пары
4	Москва	ОАО Черемушки	АО Автолог	
5	Москва	ООО Москворечье	АО Автолог	
6	Москва	АО Автолог	ООО Контакт	

Виден недостаток данного запроса: в результат попали пары близнецы (А,А) и зеркальные пары (А,В) (В,А). Эти пары необходимо исключить. Для этого используем еще один предикат, чтобы код покупателя из первой копии был больше кода покупателя из второй (можно выбрать обратное условие).

```
SELECT p1.city, p1.post_name, p2.post_name
FROM spost p1, spost p2
WHERE p1.city=p2.city
AND p1.post_kod>p2.post_kod
```

Результат запроса:

	city	post_name	post_name
1	Москва	АО Автолог	ООО Контакт
2	Москва	АО Автолог	ОАО Черемушки
3	Москва	АО Автолог	ООО Москворечье
4	Москва	ОАО Черемушки	ООО Контакт
5	Москва	ОАО Черемушки	ООО Москворечье
6	Москва	ООО Москворечье	ООО Контакт
7	СПб	АО Фонтанка	ООО Мултон

Упражнение 2. Выбрать все пары поставщиков, в названиях которых встречается форма собственности «ООО».

Замечание. Для проверки наличия фрагмента текста в названии используйте предикат подобия LIKE

3) Внешнее соединение.

Естественное или внутреннее соединение оставляют строки, совпадающие по соединяемым колонкам. В примере 1 поставщики, которые не поставляли продукцию не попали в результат запроса. Поставщики с кодами 6 и 8 не попали в результат выборки. Также в результат не попадут поставки, в которых по каким-то причинам не задан код поставщика (поставка с идентификатором 3).

Более полную картину дают внешние соединения. Внешнее соединение таблиц А и В – может быть:

- левое
- правое
- полное

Левое соединение

A LEFT OUTER JOIN B ON A.PK=B.FK

включает все строки левой таблицы А независимо от того, есть ли связанные с ней строки в таблице В.

Правое соединение

A RIGHT OUTER JOIN B ON A.PK=B.FK

включает все строки правой таблицы В независимо от того, есть ли связанные с ней строки в таблице А.

Полное соединение

A FULL OUTER JOIN B ON A.PK=B.FK

объединяет левое и правое соединения.

Все виды соединений можно представить в виде наглядных интерпретаций (рис. 4).

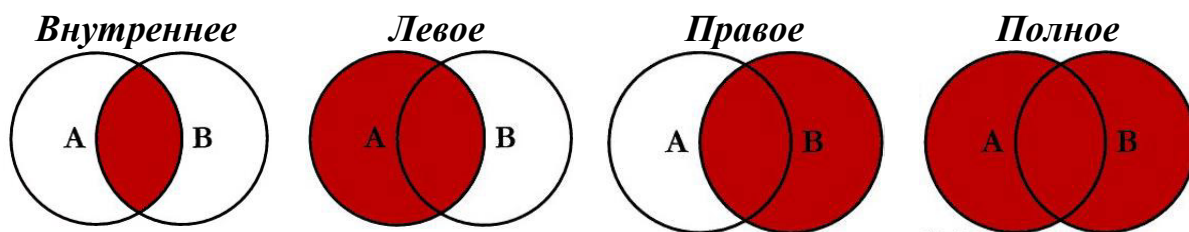


Рис.4 Наглядные интерпретации всех видов соединений

Пример 4. Вывести для всех поставщиков полную информацию об их поставках (даже, если поставок не было).

```
SELECT sp.post_name, ndoc, dat
FROM spost sp LEFT OUTER JOIN book b
ON b.post_kod=sp.post_kod
```

Будут отобраны все поставщики, независимо от того поставляли они продукцию или нет. Если поставщик не выполнял поставок, то он попадет в результат запроса, и в этой строке колонки ndoc и dat будут содержать null; а если поставки были, то число его вхождений в результат выборки равно числу его поставок.

Результат запроса:

	post_name	ndoc	dat
1	АО Автолог	28	2023-02-10
2	АО Продлог	10	2023-01-30
3	АО Продлог	19	2023-02-04
4	АО Фонтанка	(NULL)	(NULL)
5	ОАО Черемушки	26	2023-02-09
6	ООО Контакт	29	2023-02-10
7	ООО Логистик	11	2023-01-31
8	ООО Москворечье	21	2023-02-05
9	ООО Мултон	(NULL)	(NULL)

Пример 5. Вывести информацию о всех поставках с указанием наименования поставщика.

```
SELECT book_id, b.post_kod, sp.post_name, ndoc, dat
FROM spost sp RIGHT OUTER JOIN book b
ON b.post_kod=sp.post_kod
```

Будут отобраны все строки книги поставок, независимо от того, указан ли поставщик. Если поставщик не указан, то колонка post_name будет содержать null.

Результат запроса:

	book_id	post_kod	post_name	ndoc	dat
1	1	1	АО Продлог	10	2023-01-30
2	2	3	ООО Логистик	11	2023-01-31
3	3	(NULL)	(NULL)	16	2023-02-02
4	4	1	АО Продлог	19	2023-02-04
5	5	4	ООО Москворечье	21	2023-02-05
6	6	5	ОАО Черемушки	26	2023-02-09
7	7	7	АО Автолог	28	2023-02-10
8	8	2	ООО Контакт	29	2023-02-10

Пример 6. Вывести информацию о всех поставках и поставщиках

```
SELECT book_id, b.post_kod, sp.post_name, ndoc, dat
FROM spost sp FULL OUTER JOIN book b
ON b.post_kod=sp.post_kod
```

Результат объединяет данные предыдущих двух запросов:

	book_id	post_kod	post_name	ndoc	dat
1	(NULL)	(NULL)	ООО Мултон	(NULL)	(NULL)
2	(NULL)	(NULL)	АО Фонтанка	(NULL)	(NULL)
3	1	1	АО Продлог	10	2023-01-30
4	2	3	ООО Логистик	11	2023-01-31
5	3	(NULL)	(NULL)	16	2023-02-02
6	4	1	АО Продлог	19	2023-02-04
7	5	4	ООО Москворечье	21	2023-02-05
8	6	5	ОАО Черемушки	26	2023-02-09
9	7	7	АО Автолог	28	2023-02-10
10	8	2	ООО Контакт	29	2023-02-10

Упражнение 3.

В БД имеются две таблицы

Товары (STOV)

Код	Наименов.	Группа
010	Samsung 32"	Телевизоры
011	LG DF4	СВЧ печь
012	Bork A3	СВЧ печь
013	LG 41"	Телевизоры
014	Panasonic 26"	Телевизоры
Tov_kod	Tov_name	Grup

Книга продаж (BOOK)

Ид-р книги	Покупатель	Код тов.(FK)	Кол-во
1	АО Смена	010	10
2	ООО Темп	011	12
3	АО Смена	011	25
4	АО Смена	null	16
5	Null	012	8
6	ПАО Крок	013	18
Book_Id	Pokup	Tov_kod	Kol

1. Выбрать все пары товаров, относящихся к группе «телевизоры»
2. Составить операторы запроса для выполнения различных видов соединений (JOIN):

- внутреннего;
- внешнего (левого);
- внешнего (правого);
- внешнего (полного);

и привести результаты их выполнения.

Для выполнения задания создайте временные таблицы (временные таблицы существуют только в текущем сеансе запроса)

```
WITH stov(tov_kod,tov_name,grup) as
  (select '010', 'Samsung 32', 'Телевизоры'
   union
   select '011', 'LG DF4', 'СВЧ печь'
   union
   select '012', 'Bork A3', 'СВЧ печь'
   union
   select '013', 'LG 41', 'Телевизоры'
   union
   select '014', 'Panasonic 26', 'Телевизоры'),
 book (book_id,pokup,tov_kod,kol) as
  (select 1, 'АО Смена', '010',10
   union
   select 2, 'ООО Темп', '011',12
   union
   select 3, 'АО Смена', '011',25
   union
```



```
select 4, 'АО Смена', null,16
union
select 5, null, '012',8
union
select 6, 'ПАО Крок', '013',18)
SELECT ... текст оператора запроса
```

Задание

1. Выполните упражнения на учебной БД
2. Составьте похожие запросы для вашей БД